



PROJETO PARA ENSINO DE LINGUAGEM DE PROGRAMAÇÃO UTILIZANDO A PLATAFORMA RASPBERRY PI

Thiago Oliveira Rodrigues – thiagoliveira08@gmail.com

Renan Rabelo Soeiro – renan.ifce@gmail.com

Mauricio Barros de Almeida Neto – mauricio.ifce.br@gmail.com

Fillipe Diego Ferreira Carneiro – fillipediego@gmail.com

Katielle Dantas Oliveira – katielledantas@gmail.com

Rogers Guedes Feitosa Teixeira – rogerguedes.ft@gmail.com

José Alexandre Castro Bezerra Filho – j.alexandre.castro@gmail.com

Grupo de Desenvolvimento de Sist. de Telecomunicações e Embarcados - GDESTE

Avenida 13 de Maio, 2081

60040-531 – Fortaleza – CE

Carlos Maurício Jaborandy de Mattos Dourado Júnior – cmauriciojd@gmail.com

Instituto Federal de Educação, Ciência e Tecnologia do Ceará - IFCE

Avenida 13 de Maio, 2081

60040-531 – Fortaleza – CE

***Resumo:** Alguns profissionais de engenharia têm acompanhado a ascensão dos Sistemas Embarcados no que se diz respeito ao dia-a-dia. Com isso, surge a necessidade de dar continuidade a esta evolução de maneira sustentável, provendo maneiras para esse propósito. Para os problemas que surgem, a lógica de programação aparece como uma das melhores alternativas para resolvê-los. No entanto, desenvolver habilidades nesse ramo pode não ser uma tarefa muito simples. Este trabalho consiste no uso das ferramentas Scratch e Raspberry Pi no aprendizado de lógica de programação voltada para o estudo de Sistemas Embarcados.*

***Palavras-chave:** Raspberry Pi, Scratch, Sistemas Embarcados, Lógica de Programação*



1. INTRODUÇÃO

O problema central das instituições de ensino superior brasileiras é o alto índice de evasão. Os cursos na área de Ciências, Matemática e Computação tem uma taxa em torno de 28% de evasão. Esse valor é considerado alto, já que a média nacional é de 23%, segundo dados divulgados pelo INEP em 2005. Nota-se que as disciplinas relacionadas a algoritmos ou linguagens de programação são umas das razões pela qual há a evasão e reprovação nas primeiras fases dos cursos de informática. (RICHARDSON, 2012)

O objetivo dessa disciplina é fazer o aluno compreender a forma de “raciocínio” do computador e dominar as técnicas para solucionar problemas propostos. Estas soluções pode ser um programa de computador ou só um pseudocódigo – o algoritmo. Um algoritmo é um conjunto finito de comandos que resolverão um problema proposto. (FILHO, 2007)

Existe uma dificuldade de adaptação dos alunos em desenvolver raciocínio lógico, pois estão acostumados a decorar o conteúdo e a falta de motivação do aluno gerada pelo despreparo, podem ser citadas como motivos para a reprovação ou evasão do aluno.

A proposta deste artigo consiste em apresentar um projeto para a introdução do estudo de linguagens de programação, e até mesmo de hardware, a jovens entusiastas na faixa etária de 12 à 17 anos de idade e estudantes dos cursos de engenharia. Aproveitando a motivação e a busca por novidade de crianças e adolescentes para ensiná-los, de forma prática e divertida, a ter um raciocínio lógico e despertando o interesse dos mesmos nas áreas de engenharia e suas tecnologias.

2. OBJETIVOS

"Programação é importante pelo mercado de trabalho, que pede muito isso, mas nosso objetivo principal não é esse. Queremos desenvolver certas habilidades que a programação exige dos alunos, como raciocínio lógico, causa e consequência, raciocínio matemático, linguística... o que está inserido no dia a dia" afirma Mateus Ferreira - Professor de Informática do colégio Vértice.

O projeto apresentado tem como principal objetivo despertar o raciocínio lógico de alunos e jovens, incentivando-os e introduzindo-os no estudo das linguagens de programação para que futuramente estes não sintam dificuldades ao participar de disciplinas que envolvam raciocínio lógico ou algum tipo de linguagem de programação.

Além disso, o projeto tem como objetivo:

- Incentivar o ensino de novas linguagens;
- Incentivar o uso de novas metodologias aplicadas ao ensino;
- Treinar alunos a utilizarem a ferramenta proposta;
- Ensinar o uso da Raspberry Pi;
- Ensinar o uso do Scratch.
- Criar uma ferramenta que atue na pedagogia infantil e juvenil;
- Utilizar a tecnologia para o ensino.

3. FUNDAMENTAÇÃO TEÓRICA

O início da ciência da Lógica encontra-se na antiga Grécia. As polêmicas geradas pela teoria de Parmênides e os famosos argumentos de Zenão, que negavam a realidade do movimento fazendo um uso indevido do princípio da não contradição, contribuíram para a distinção dos conceitos, para se ver a necessidade de argumentar com clareza, mediante demonstrações rigorosas, e assim responder às objeções dos adversários.

Mais tarde, as sutilezas dos sofistas, que reduziam todo o saber à arte de convencer pelas palavras, levaram Sócrates a defender o valor dos conceitos e tentar defini-los com precisão. Assim a Lógica como ciência vai se formando pouco a pouco, principalmente com Sócrates e Platão.

Os primeiros passos foram dados com Seymour Papert e seu conceito de construcionismo. Ele desenvolveu uma nova abordagem para “pensar, aprender e resolver problemas” geométricos. Esta abordagem, que ele chamou de geometria da tartaruga, se utiliza de ferramentas computacionais. Da mesma maneira que as abordagens lógicas de Euclides e abordagem algébrica de Descartes, a abordagem computacional de Papert é uma ferramenta muito poderosa para “pensar, aprender e resolver problemas”, não só de geometria, mas, também, de outras áreas do conhecimento.

A criação de novas formas de inovar no ensino da programação não se limitou a linguagem LOGO, ou linguagem da tartaruga, e com o passar do tempo tais ferramentas ganharam uma nova maneira de interação com os usuários. Um exemplo dessas novas ferramentas é o Scratch. O Scratch é um programa que permite ensinar conceitos de programação para crianças e permite associar esse conhecimento à criação de projetos multidisciplinares. A Scratchboard ou PicoBorad, uma maneira para você fazer projectos Scratch responder a coisas que acontecem no mundo fora de seu computador. (CAETANO, 2011)

Criar algo a partir de um raciocínio é construtivo, segundo a psicóloga Luciana Ruffo, do Núcleo de Pesquisa em Psicologia e Informática da PUC de São Paulo. “Aprender a programar significa um exercício para o cérebro, principalmente quando envolve linguagens diferentes, como na aprendizagem de um novo idioma. Além disso, desenvolve algumas habilidades lógicas e de raciocínio que podem ser úteis”, afirma a psicóloga.

Incentivar o estudo de linguagens de programação durante o período do ensino fundamental e médio pode ajudar a desenvolver habilidades que trará benefícios para os estudantes, porém esse estudo deve ser uma opção para o aluno, pois a imposição desse estudo será somente uma matéria a mais e não trará benefícios ao aluno. “De alguma forma, ser forçado a aprender algo que não faz o menor sentido para a pessoa é complicado”, indagou Luciana.

Para que a aplicação do estudo não se torne cansativa e pouca atrativa para os alunos, principalmente para aqueles de menor idade, é necessário conseguir um nível de interatividade que torne mais fácil a assimilação do conteúdo, podendo assim desenvolver mais rapidamente suas habilidades.

Posteriormente será mostrado qual material e tecnologia que será adotado para despertar o interesse e manter a atenção dos estudantes na aplicação dos minicursos.

4. REVISÃO BIBLIOGRÁFICA

4.1. Scratch

O Scratch, projeto do grupo Lifelong Kindergarten no Media Lab do MIT, a escolha dessa linguagem dá-se pela sua forma de programação onde não é necessário digitar linhas de código. Sua programação é feita através de blocos de funções onde o usuário cria combinações desses blocos como se estivesse montando um quebra-cabeça.

O Scratch é uma linguagem gráfica de programação inspirada nas linguagens Logo e Squeak, mas que pretende ser mais simples, fácil de utilizar e mais intuitiva. Ela foi partilhada com o mundo em 15 de maio de 2007. É um software gratuito, possui uma IDE onde não é preciso digitar funções, endereços, etc. Seu objetivo primário é facilitar a introdução de conceitos de matemática e de computação, enquanto também induzindo o pensamento criativo, o raciocínio sistemático e o trabalho colaborativo. (SCRATCH, 2013)

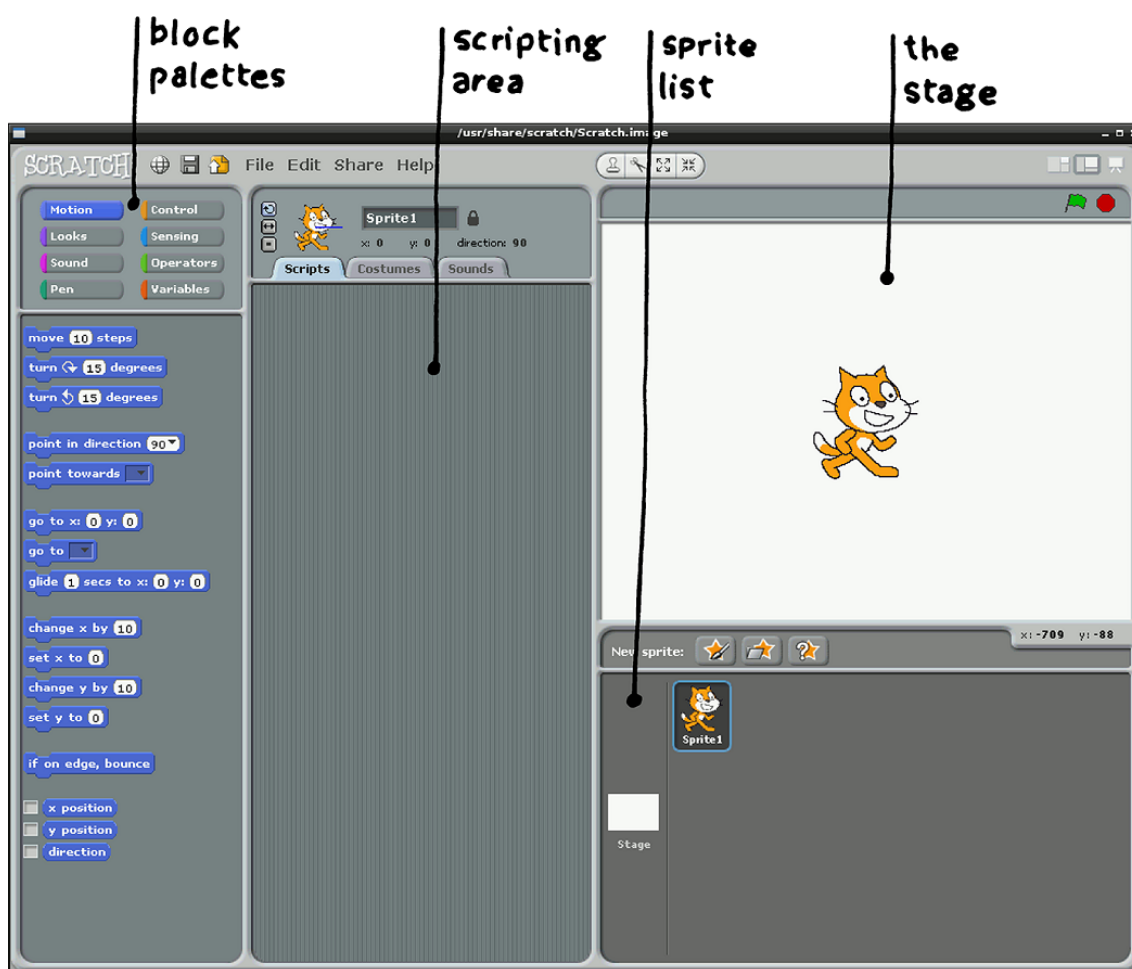


Figura 1 - Ambiente Scratch: tudo está contido em uma janela, com subpainéis para as paletas de blocos, área de scripting e customização, lista de estágios e palco.

Com o Scratch é possível misturar diferentes recursos (gráficos, fotos, músicas, sons) de uma forma bastante criativa, daí provém a origem se deu nome. O termo Scratch provém da técnica de scratching utilizada pelos Disco-Jockeys do Hip-Hop que giram os discos de vinil com as mãos para frente e para trás, fazendo misturas musicais criativas.

O Scratch traz os conceitos de informática e lógica de uma forma diferenciada para seus usuários, pois a programação do Scratch é feita por meio de blocos de comandos que são encaixados uns aos outros, formando uma sequência desejada. Esses blocos somente se encaixam de uma única forma, não ocorrendo, assim, erros de sintaxe.

4.2. Raspberry Pi

A Raspberry Pi, desenvolvido pela fundação Raspberry Pi, tem como principal finalidade o auxílio ao estudo de programação e tecnologia fornecendo uma plataforma que engloba as principais tecnologias utilizadas atualmente. Essas tecnologias são: saídas de áudio e vídeo, portas USB, armazenamento onboard e rede onboard, além de apresentar suporte a outras tecnologias como o GSM.

Todo o hardware é integrado em uma única placa. Um dos objetivos principais é de estimular o ensino de ciência da computação básica em escolas. Este minicomputador contém um processador ARM da família 11, com clock de 700 Mhz, uma memória SDRAM de 512 Mbytes, duas portas USB 2.0, uma de vídeo HDMI, uma saída de áudio para conector de 3,5 mm, um slot para cartão SD, um slot para rede Ethernet do padrão RJ45, interfaces de comunicação UART, I²C e SPI. Esta plataforma permite a instalação de sistemas operacionais baseados em linux, tais como: Debian GNU/Linux, Fedora, Arch Linux, Raspbian, RISC OS, etc. Segue na figura 3 a plataforma Raspiberry Pi modelo B e ao lado suas especificações de forma identificada.

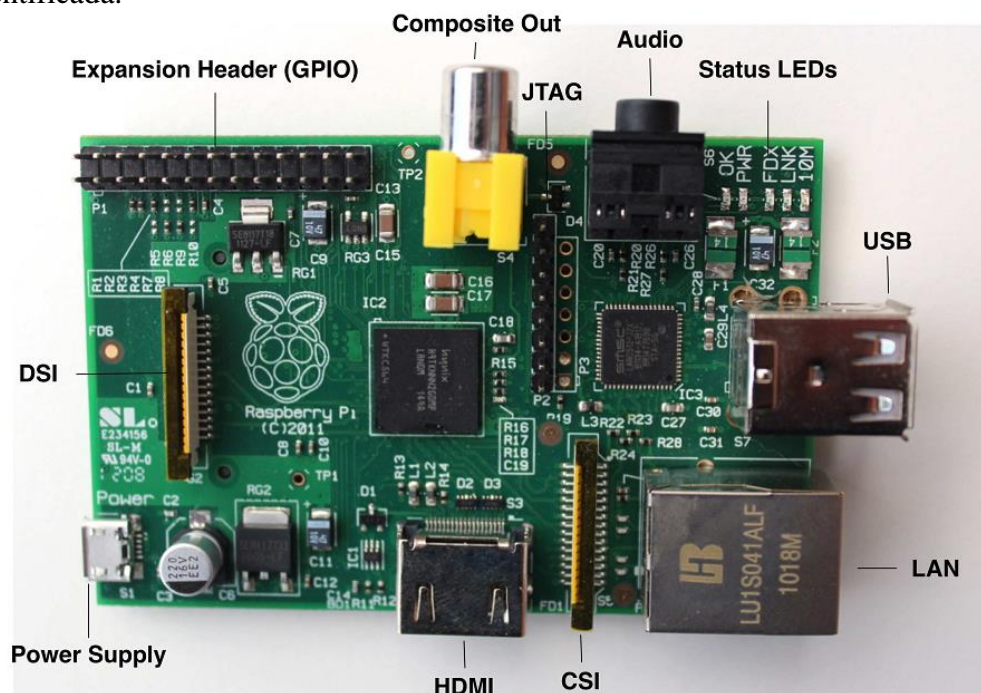


Figura 2 - Imagem frontal do modelo B

4.3. Lógica de programação

O objetivo da lógica de programação é fazer as pessoas compreenderem a forma de “raciocínio” do Computador e dominar as técnicas para solucionar problemas propostos. Estas soluções Podem ser um programa de computador ou só um pseudocódigo – o algoritmo. Um Algoritmo é um conjunto finito de comandos que resolverão um problema proposto. (CAETANO, 2011)

5. METODOLOGIA

Para este projeto será usado o computador Raspberry Pi composto com o ambiente de programação Scratch formando uma plataforma interativa específica para o aprendizado de linguagens de programação.

A escolha dessas plataformas dá-se pela aceitação na comunidade científica e por serem plataformas já testadas e consolidadas no mercado, além de apresentarem um bom custo benefício.

Com a plataforma para ensino montada, será promovido cursos que ensinaram os alunos a utilizarem a ferramenta para que posteriormente possa ser aplicado o estudo das linguagens de programação.

6. RESULTADOS

“Além do conhecimento tecnológico – a tendência é que a programação esteja presente em todas as áreas -, o estudante aprende a raciocinar de uma forma diferente, que pode ajudá-lo na outras matérias.”, conclui o Prof. Mateus Ferreira.

Ao final das etapas deste trabalho vislumbra-se a necessidade dos alunos porem em prática a teoria vista em sala pela necessidade de ver seu conhecimento gerar algo produtivo.

O principal objetivo do trabalho é além de tornar os alunos mais aptos aos ambientes de programação é desenvolver suas habilidades de raciocínio lógico e poder de decisão com isso ajudando-os a aplicar seus novos conhecimentos em outras matérias e no dia-a-dia em geral.

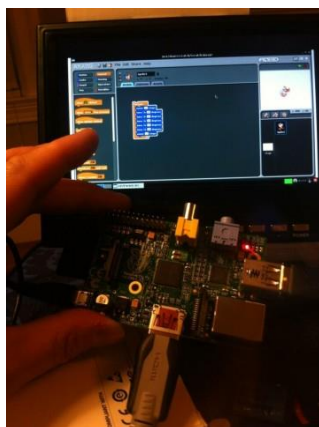


Figura 3 - Scratch sendo utilizada junto com a Raspberry Pi.

7. CONCLUSÃO

O aprendizado em lógica computacional é de fundamental importância para os jovens futuros engenheiros e profissionais do ramo.

Pode-se concluir que as ferramentas neste artigo apresentadas são bastante eficientes no ensino da lógica de programação, devido ao contato com ambientes pouco complexos e intuitivos.

Agradecimentos

Agradecemos ao IFCE (Instituto Federal de Educação, Ciência e Tecnologia do Ceará) e ao laboratório GDEST (Grupo de Desenvolvimento de Sistemas de Telecomunicações e Sistemas Embarcados), os quais me deram suporte no desenvolvimento desse artigo.

8. REFERÊNCIAS / CITAÇÕES

Livros:

FARRER, Harry, Algoritmos Estruturados, Editora LTC, 1989, Rio de Janeiro.

RICHARDSON, Matt; WALLACE, Shawn. Getting Started with Raspberry Pi, O'Reilly, 2012.

SCHMIDT, Maik; CARTER, Jaquelyn. Raspberry Pi – A Quick-Start Guide, Texas, Pragmatic Bookshelf, 2012.

CAETANO, Daniel. Lógica de Programação Para Engenharia, Unidade 1: Introdução à Disciplina. São Paulo: 2011.

FONSECA FILHO, Clézio. História da Computação: O caminho do pensamento e da tecnologia, 1. ed. Porto Alegre: EDIPUCRS, 2007. 205 p, il.

FORBELLONE, André Luiz Villar; EBERSPÄCHER, Henri Frederico. Lógica de Programação: A construção de algoritmos e estruturas de dados. 3. ed. São Paulo: Pearson Prentice Hall, 2005. 220 p, il.

Monografias, dissertações e teses:

GUEDES, R. B. M.; UNIVERSIDADE FEDERAL DE PERNAMBUCO, Centro de Ciências Exatas e da Natureza. Inteligência Computacional: Métodos Procedimentais para Pensar, Aprender e Resolver Problemas, 1998. 202p, il. Dissertação (Mestrado).



Trabalhos em eventos:

CABRAL, M. I. C. et al. Perfil dos cursos de computação e informática no Brasil, XXVII Congresso da SBC – XV WEI, Rio de Janeiro, 2007.

PEREIRA, P. S. et al. Análise do Scratch como ferramenta de auxílio ao ensino de programação de computadores. Anais: XL – Congresso Brasileiro de Educação em Engenharia. Belém: UFPA, 2012.

POCRIFKA, D. H. Linguagem Logo e a construção do conhecimento. Anais: IX – Congresso Nacional de Educação. Curitiba: PUCPR, 2009.

Internet:

SCRATCH, MIT Media Lab. Disponível em < <http://scratch.mit.edu/> >. Acesso em: 09 mai 2014.

SCRATCH 4 ARDUÍNO, S4A. Disponível em < <http://s4a.cat/> >. Acesso em: 09 mai 2014.

DANIEL, Marcelo. **Pequenos Criadores**, São Paulo, Revista Educação, p. 78, Ed. nov 2013. Disponível em < <http://colégiovertice.com.br/wp-content/uploads/2013/11/Mat%C3%A9ria-Revista-Educa%C3%A7%C3%A3o.pdf> >. Acesso em: 09 mai 2014.

SCRATCH. **ABOUT Scratch (Scratch Documentation Site)**. Disponível em: < <http://scratch.mit.edu/about/> >. Acesso em: 10 junho 2013.

Citilab. **Scratch for Arduino**. Disponível em: <<http://seaside.citilab.eu/scratch/arduino>> Acesso em: 25 mai. 2012.

Massachusetts Institute of Technology (MIT). **Scratch Sensor Board**. Disponível em: <http://info.scratch.mit.edu/Support/Sensor_Boards> Acesso em: 02 jun. 2012.

Scratch Brasil. **O que é o Scratch?** Disponível em: <<http://scratchbrasil.wordpress.com/>> Acesso em: 08 jun. 2012.

[18] Adafruit. **Raspberry Pi Running Scratch**. Disponível em: <<http://www.adafruit.com/blog/2013/04/12/59738/>> Acesso em: 09 mai. 2014.



PROJECT FOR TEACHING OF PROGRAMMING LANGUAGE USING THE PLATFORM RASPBERRY PI

***Abstract:** Some engineering professionals have accompanied the rise of Embedded Systems as regards the day-to-day. With this comes the need to continue this development in a sustainable manner, providing ways for this purpose. To the problems that arise, logic programming appears as one of the best alternatives to solve them. However, developing skills in this business can't be a simple task. This work is the use of Scratch and Raspberry Pi tools in logic programming focused on the study of learning embedded systems.*

***Key-words:** Raspberry Pi, Scratch, Embedded Systems, Logic Programming*